

BAB 2

TINJAUAN PUSTAKA

1.1 Sistem Pendukung Keputusan

Sebuah sistem pendukung keputusan (DSS) merupakan Sistem yang dapat menawarkan jawaban atas masalah dalam keadaan semi dan tidak terstruktur. Sistem pendukung keputusan (DSS) adalah perangkat lunak komputer yang memiliki akses ke database dan domain aplikasi yang disediakan oleh model analisis keputusan. Tujuannya adalah untuk membantu pengambil keputusan dalam membuat keputusan yang tepat dengan terstruktur maupun tidak. Keseluruhan struktur organisasi mencakup sistem pengambilan keputusan secara keseluruhan. Jadi, sistem organisasi paling tidak terdiri dari sistem informasi, sistem manajemen, dan sistem operasional. (Burhanuddin & Dini, 2017).

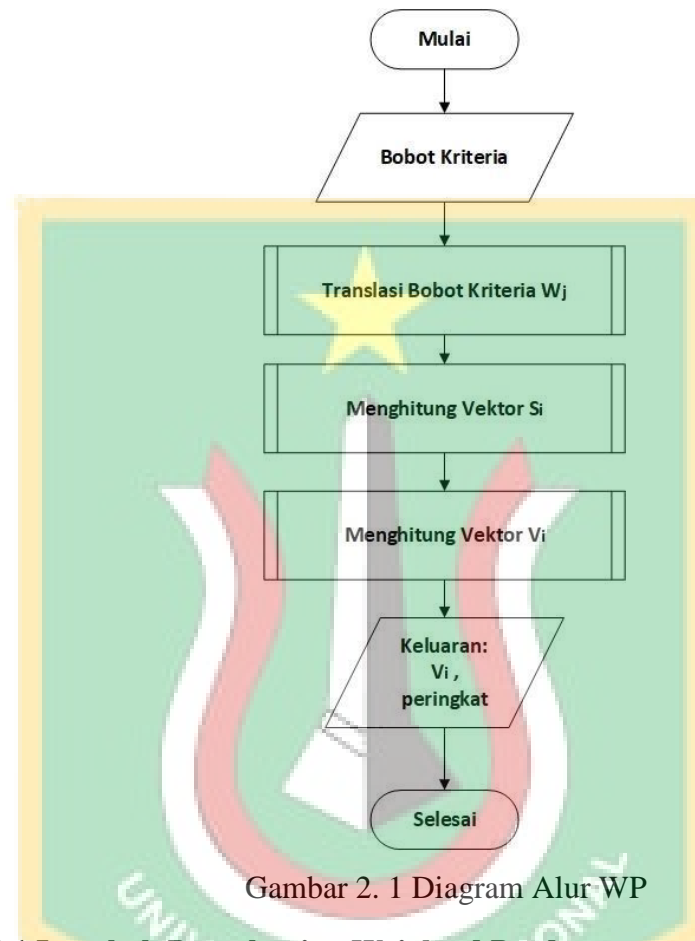
1.2 XAMPP

XAMPP adalah perangkat lunak bebas dan di dukung dengan banyak system operasi, artinya yaitu XAMPP ialah kompilasi dari beberapa program. XAMPP adalah tool yang menyediakan paket aplikasi kedalam satu buah paket. dalam menginstall XAMPP maka tidak perlu lagi mengisntal dan mengkonfigurasi manual web server Apache, PHP dan MySQL karena sudah ada di dalamnya. XAMPP juga merupakan suatu perangkat lunak yang bersifat open source yang merupakan pengembangan dari LAMP (Linux, Apache, MySQL, PHP dan Perl) (Martine, 2018).

1.3 Penjelasan Weighted Product

Metode Weighted Product dapat didefinisikan sebagai keputusan analisis multi kriteria yang banyak digunakan dan merupakan sebuah metode pengambilan keputusan multi kriteria, yang masuk kedalam metode *Fuzzy Multiple Attribute Decision Making* (FMADM), Weighted Product juga dapat diartikan sebagai himpunan berhingga dari alternatif keputusan yang dijelaskan oleh istilah beberapa kriteria keputusan. Diagram alur

Weighted Product dapat ditunjukkan pada Gambar 2.1 (Burhanuddin & Dini, 2017).



2.3.1 Langkah Penyelesaian Weighted Product

Metode Weighted Product adalah suatu metode pada pengambilan sebuah Keputusan meningkat untuk menghubungkan nilai dasar, di mana nilai setiap standar awalnya harus dinaikkan ke kekuatan bobot aturan yang relevan (Junifa et al., 2019).

A. Nomalisasi atau Perbaikan bobot

$$W_j = \frac{W_j}{\sum W_j}$$

Menormalkan atau mengoreksi bobot agar dapat memberikan nilai $W_j = 1$, di mana $j = 1, 2, \dots, n$ adalah

jumlah opsi lain dan W_j adalah jumlah absolut dari nilai tertimbang.

A. Menentukan Nilai Vektor S

$$S_i = \prod_{j=1}^n X \frac{W_j}{i_j} \quad w_j \dots \dots \dots$$

menghitung semua kriteria dengan bobot yang dinormalisasi atau ditingkatkan dengan tingkat positif untuk kriteria manfaat (benefit) dan kriteria dengan peringkat negatif untuk kriteria biaya (cost), memastikan nilai vektor S. S adalah preferensi untuk kriteria, x adalah nilai kriteria, dan n adalah banyaknya kriteria.

Memastikan nilai vektor S melibatkan mengalikan setiap kriteria dengan hasil alternatif, seperti perbaikan bobot atau normalisasi, yang memiliki peringkat positif untuk kriteria menguntungkan (benefit) dan peringkat negatif untuk kriteria (cost) yang tidak menguntungkan, di mana S nilai adalah preferensi kriteria, X adalah kriteria nilai, dan nilai n adalah jumlah kriteria.

B. Menentukan Nilai Vektor V

$$V_i = \frac{\prod_{j=1}^n X \frac{w_j}{i_j}}{\prod_{j=1}^n (X \frac{W_j}{j})^{w_j}}$$

Mencari nilai hasil vektor V, preferensi alternatif yang akhirnya digunakan untuk meranking setiap jumlah nilai vektor S, total dari semua nilai vektor S.

1.4 Penjelasan SMART

SMART merupakan metode yang menggunakan adaptif linier. Dimana Nilai setiap alternatif diprediksi oleh SMART menggunakan model adaptif linier. Karena kemudahannya memenuhi kebutuhan pembuat keputusan dan cara menganalisis respon, oleh karena itu metode SMART banyak digunakan. Transparansi adalah analisis terbaik karena menawarkan pemahaman menyeluruh tentang masalah dan dapat diterima oleh pembuat keputusan. Dengan menggunakan skala 0 hingga 1, pembobotan SMART memudahkan penghitungan dan perbandingan hasil untuk setiap alternatif. (Rahayu et al., 2021).

2.4.1 Tahapan Penyelesaian SMART

A. Menentukan Kriteria

Menentukan kriteria yang digunakan dalam memecahkan permasalahan dalam mengambil keputusan. dalam menetapkan kriteria-kriteria mana yang digunakan untuk sistem pengambilan keputusan yang dijalankan, diperlukannya data-data dari pembuat keputusan atau pihak yang bertanggung jawab terhadap permasalahan yang terjadi.

B. Menentukan Bobot Kriteria

Menetapkan bobot kriteria pada setiap kriteria dengan menggunakan nilai dari 1 hingga 100 pada setiap kriteria dengan prioritas terpenting.

C. Normalisasi Bobot Kriteria

Menghitung normalisasi bobot pada setiap kriteria dengan cara membandingkan nilai bobot kriteria dengan jumlah keseluruhan bobot kriteria.

D. Memberikan Nilai Parameter untuk Tiap Kriteria

Memberi nilai kriteria pada setiap alternatif, dalam pemberian nilai kriteria pada setiap alternatif dapat berupa data kuantitatif (angka) atau data kualitatif. Misalnya nilai kriteria harga tidak diragukan lagi berupa data kuantitatif, dan nilai kriteria kelayakan sarana dapat dipresentasikan melalui data kualitatif (sangat lengkap, lengkap, kurang lengkap). Untuk menentukan parameter nilai kriteria, kita harus mengubah nilai kriteria kualitatif menjadi nilai kriteria kuantitatif, misalkan sangat lengkap memiliki nilai 5, lengkap memiliki 4 dan tidak lengkap memiliki nilai 3.

E. Menentukan Nilai Utility

Mengkonversi nilai kriteria dengan setiap kriteria untuk menentukan nilai utility, yang kemudian menjadi nilai standar data kriteria. Sifat kriteria itu sendiri menentukan nilai utility.

Kriteria Biaya (Cost Criteria)

Kriteria “nilai yang lebih kecil lebih diinginkan” kriteria tersebut banyak digunakan seperti biaya yang harus dikeluarkan misalnya (kriteria penetapan harga, kriteria penggunaan bahan bakar per kilometer saat pembelian mobil, periode pengembalian modal perusahaan, kriteria lamanya pengiriman) dari kriteria yang digunakan tersebut dihitung dengan persamaan :

$$U_i(a_i) = 100 \frac{(C_{max} - C_{out})}{C_{max} - C_{min}} \%$$

Keterangan

$u_i(a_i)$: nilai utility kriteria ke-i untuk alternatif ke-i

c_{max} : nilai dari kriteria maksimal

c_{min} : nilai dari kriteria minimal

c_{out} : nilai kriteria ke- i

Kriteria Keuntungan (Benefit Criteria)

Kriteria Keuntungan yaitu kriteria yang bersifat “lebih diinginkan lebih besar”, pada kriteria ini biasanya berbentuk keuntungan (misalnya kriteria kapasitas tangki untuk pembelian mobil, kriteria kualitas dan lain-lain).

F. Menentukan Nilai Akhir

Nilai yang telah di dapat dari normalisasi nilai data kriteria data baku dikalikan dengan nilai normalisasi bobot kriteria untuk mendapatkan hasil akhir.

$$U(a_i) = \sum_{j=1}^m W_j U_j(a_i), i = 1, 2, \dots, m$$

Keterangan:










$u(a_i)$: nilai total untuk alternatif ke- i

w_j : nilai bobot kriteria ke- j yang sudah ternormalisasi

$u_j(a_i)$: nilai utility kriteria ke- j untuk alternatif ke- i

1.5 Flowchart

Flowchart merupakan suatu bagan yang menggunakan simbol-simbol tertentu untuk menunjukkan hubungan antara satu proses atau instruksi dan proses lainnya dalam program yang dikembangkan serta urutan rinci di mana setiap proses terjadi (Wibawanto 2017). Pada tabel 2.1 adalah simbol-simbol yang digunakan untuk menggambarkan suatu flowchart :

SIMBOL	NAMA	FUNGSI
	TERMINATOR	Permulaan / akhir program
	GARIS ALIR (FLOW LINE)	Arah aliran program
	PREPARATION	Proses inisialisasi/pemberian harga awal
	PROSES	Proses perhitungan/proses pengolahan data
	INPUT/OUTPUT DATA	Proses input/output data, parameter, informasi
	PREDEFINED PROCESS (SUB PROGRAM)	Permulaan sub program/proses menjalankan sub program
	DECISION	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	ON PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada satu halaman
	OFF PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda

Tabel 2. 1 Simbol Flowchart

1.6 HTML (Hypertext Markup Language)

HyperText Markup Language atau HTML, adalah bahasa pemrograman standar yang digunakan untuk membangun sebuah halaman web, yang selanjutnya dapat diakses untuk menampilkan berbagai jenis informasi di dalam sebuah penjelajah web Internet (*Browser*). HTML juga dapat digunakan sebagai link link antara file-file pada situs atau untuk komputer dengan menggunakan localhost, atau link yang digunakan untuk menghubungkan antar situs dalam dunia internet.







1.7 Unified Modelling Language (UML)

UML (*Unified Modelling Language*) merupakan salah satu alat bantu yang banyak digunakan dalam dunia pengembangan sistem berorientasi obyek. Hal ini disebabkan karena UML menawarkan bahasa pemodelan

visual yang memungkinkan perancang sistem untuk menyusun cetak biru ide mereka dalam format standar yang mudah dipahami dan dilengkapi dengan metode yang efisien untuk berbagi dan mengkomunikasikan rancangan mereka dengan yang lainnya (Martine, 2018).

2.8.1 Usecase Diagram

Use case adalah deskripsi operasi sistem dari sudut pandang pengguna. Use case menggambarkan dan menjelaskan suatu interaksi umum antara pengguna sistem dan sistem yang dijalani. Dengan menceritakan bagaimana sebuah sistem digunakan (Yudistira & Sari, 2020). Adapun Simbol dalam membuat usecase diagram ditunjukkan pada tabel 2.2.

No	Nama	Simbol	Fungsi
1	System Boundary		Menyatakan batasan sistem dalam relasi dengan aktor-aktor yang menggunakannya dan fitur-fitur yang harus disediakan.
2	Actor		Segala sesuatu yang perlu berinteraksi dengan sistem untuk pertukaran informasi, bisa merupakan manusia, sistem atau device.
3	Use Case		Mengidentifikasi fitur kunci dari sistem, tanpa sistem ini tidak akan bisa memenuhi permintaan user.
4	Asoslotion		Mengidentifikasi interaksi antara setiap aktor tertentu dengan setiap use case tertentu.
5	Include		Mengidentifikasi kelakuan yang harus terpenuhi agar sebuah event dapat terjadi, kondisi ini adalah hubungan dua use case dimana yang satu memanggil yang lain.
6	Extend		Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber secara eksplisit.

Tabel 2. 2 Simbol Use Case Diagram

2.8.2 Class Diagram

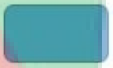




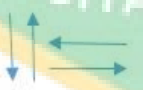
Class diagram merupakan model statis yang menggambarkan struktur dan deskripsi class serta hubungan antara class. Class diagram mirip dengan ER-Diagram pada perancangan basis datanya, perbedaan adalah pada ER-diagram tidak memiliki operasi/methode tetapi hanya atribut. Class terdiri dari nama kelas, atribut, dan operasi/methode. Adapun Simbol dalam class diagram ditunjukkan pada tabel 2.3.

No	Nama	Simbol	Fungsi
1	Class		Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
2	Association		Relasi antar kelas dengan makna umum, asosiasi juga biasanya juga disertai dengan multiplicity.
3	Directed Association		Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity.
4	Generalisasi		Relasi antar kelas dengan makna generalisasi-spesialisasi.
5	Dependency		Relasi antar kelas dengan makna ketergantungan antar kelas.
6	Aggregation		Relasi antar kelas dengan makna semua-bagian (whole-part).

Tabel 2. 3 Class Diagram.

2.8.3 Activity Diagram

Diagram activity berguna untuk menggambarkan alur kerja atau aktifitas yang dilakukan system bukan aktor. Hal ini berfokus pada aktifitas-aktifitas yang saling terkait yang terjadi dalam proses tunggal. Activity diagram dibagi menjadi beberapa jalur. Itu berguna untuk menunjukan objek-objek yang bertanggung jawab untuk setiap aktifitas. (Fakhriyah, 2021). Adapun Simbol-Simbol dalam menggambarkan diagram activity pada tabel 2.4.

No	Nama	Simbol	Fungsi
1	Activity		Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2	Action		State dari sistem yang mencerminkan eksekusi dari suatu aksi.
3	Initial Node		Bagaimana objek dibentuk atau diawali.
4	Activity Final Node		Bagaimana objek dibentuk dan diakhiri.
5	Decision		Digunakan untuk menggambarkan suatu keputusan/tindakan yang harus diambil pada kondisi tertentu.
6	Line Connector		Digunakan untuk menghubungkan satu simbol dengan simbol lainnya.

Tabel 2. 4 Simbol activity Diagram

1.8 PHP (Hyper Text Preprocessor)

PHP ialah Bahasa server-side –scripting yang menyatu dengan HTML untuk membangun halaman web yang dinamis. Karena PHP adalah server-side-scripting oleh karena itu sintaks dan perintah-perintah PHP akan diproses diserver kemudian hasil yang diperoleh akan dikirimkan ke browser dalam format HTML (Munawar et al., 2021).

PHP bersifat open source dan tersedia gratis (free). PHP diterbitkan dalam lisensi PHP License, sedikit berbeda dari lisensi GNU General Public License (GPL) yang banyak digunakan dalam proyek Open Source.

1.9 Data Base

Basis data dapat dikatakan sebagai sekelompok data yang diatur secara sistematis dan disimpan didalam komputer, basis data dapat digunakan oleh program atau aplikasi untuk memproses atau mengubah untuk menghasilkan suatu informasi. Basis data didefinisikan dengan meliputi spesifikasi berupa tipe data, struktur data dan juga batasan-batasan pada data yang selanjutnya akan disimpan (Yudistira & Sari, 2020).

1.10 MySQL

MySQL adalah perangkat lunak yang digunakan dalam membangun database yang sering digunakan pada linux. MySQL merupakan software open source yang berarti free untuk digunakan. Selain di lingkungan linux, MySQL juga tersedia pada windows. Dapat disimpulkan, MySQL yaitu sebuah sistem database server yang digunakan untuk membangun aplikasi yang bersifat open source dan dapat digunakan diberbagai platform (Martine, 2018).

